Maliciously Manipulating a Robotic Swarm

Ian Sargeant and Allan Tomlinson
Information Security Group, Royal Holloway, University of London
Egham Hill, Egham. Surrey. TW20 0EX. U.K.

Abstract— *Most contemporary research in the field of robotic swarms assumes a benign operational environment. In our work we assume a hostile environment. We consider how swarms might be attacked and begin with a review of robotic swarm taxonomies. We then consider how a generic swarm might be attacked, and how attacks on swarms might be investigated. We conclude by presenting results of simulations of attacks that have been undertaken against swarms, based the robotic swarm taxonomies.*

Keywords: Robotic Swarms, Security, Attacks, Simulations

1. Introduction

Considerable research has been undertaken into the applications [1]–[3] and implementations of robotic swarms [4]–[8]. Much of this work assumes trust between the interacting swarm elements and trust of their operating environment. Consequently most physical implementations are realised in relatively benign conditions. Our concern, however, is with robotic swarms that might operate in hostile environments.

The aim of this paper is to demonstrate that robotic swarms can be subject to dedicated attacks, with an adversary exploiting the unique characteristics of the swarm.

To do this, we clearly need to understand the unique characteristics of robotic swarms that an adversary may exploit. We therefore review previously proposed robotic swarm taxonomies [9]–[13] to allow us to determine where the vulnerabilities in various types of swarm might lie.

We also need to understand the adversary's capabilities and the classes of vulnerabilities that an attacker may exploit. In the latter, we concentrate on vulnerabilities that are unique to swarm robotic systems. In particular, our aim is to be able to manipulate the swarm to alter it's behaviour.

In order to study how attacks may be used to manipulate the swarm, we use Netlogo simulations. This allows us to investigate, firstly if the swarm can be manipulated, and if so what resources are needed to carry out the attack.

1.1 Introduction to Swarm Robotics

Swarm robotic research investigates the potential uses and benefits of autonomous multi-robot systems, often taking inspiration from nature, such as swarms of bees, colonies of ants and schools of fish. The philosophy behind a robotic swarm is that a large number of relatively basic robots will work and interact with each other, and the environment [14], so as to complete a predetermined goal. It is generally agreed that a robotic swarm should have the following characteristics [14]–[18]:

- Autonomy
- Decentralised control
- Large number of members
- Collective emergent behaviour
- Local sensing
- Local communications
- Resilience to failures within the robotic swarm
- Scalable in size

In our research, the key aspects of robotic swarms are those that make them unique when compared to current information systems. In particular our interest is in the emergent behaviour of the swarm. Robotic swarms are, in theory, autonomous groups of simple swarm elements that operate remotely, with no deterministic program. Each swarm element follows a simple set of rules, interacting only locally, in order to achieve a pre-determined goal. Thus, the swarm behaviour emerges from the simple rules and local interaction. The extent to which this emergent behaviour may be disrupted is the problem that we are investigating.

At the moment, the study of swarm robotics is generally undertaken within a research environment, with proposed applications for the use of robotic swarms often being theoretical. However, the proposed numerous and diverse uses of robotic swarms, along with the the potential benefits that could be realised, drives the majority of the research. Our motivation is to understand the vulnerabilities in these systems, and how they may be mitigated, before systems are widely deployed.

There are proposed applications of robotic swarms in both civilian and military applications.

Typical civilian applications include [2], [3]:

- Maintenance tasks
- Communications provision
- Emergency response
- Use in space flights and exploration
- Use in medical procedures

Typical military applications include [19], [20]:

- Military communications
- Underwater mine clearance
- Landmine detection
- Situational Awareness

ISBN: 1-60132-433-2, CSREA Press ©

1.2 Introduction to the Problem

In our work, we assume that the swarm is operating in a hostile environment. And we assume that an adversary has the opportunity, and ability, to attempt to manipulate the emergent behaviour of the swarm. The goal of the adversary might be to prevent the swarm from achieving its goal, or it may be simply to slow the progress of the swarm. The latter might be an easier attack to undertake.

The overarching problem is therefore to determine whether the expected behaviour of a robotic swarm can be altered by an attacker, and if so, to what extent?

2. Taxonomy of Robotic Swarms

Many swarm designs have been proposed and although each may have specific vulnerabilities our goal is to identify generic vulnerabilities. Hence we need to be able to classify swarms somehow, and identify common characteristics that may be vulnerable to common types of attack.

There are various proposals defining robotic swarm taxonomies in the literature. These taxonomies are based on several factors, such as physical design considerations and collective behaviours [9]. Typical examples being the size of the swarm, communications media, coordination, control, design approach and processing abilities, and collective behaviours, such as foraging or construction [10]–[13].

Other proposals [21], [22] suggest that the attributes for consideration should be the highly dependent features of group architecture, how members deal with resource conflicts, how a collective learns and adapts to a task, how cooperation is motivated and how planning is addressed. Indeed, certain taxonomies [22] also consider the environment in which the entities are operating.

The following is an overview of the swarm taxonomies proposed to date.

2.1 Method Based Taxonomy

Brambilla et al. [9] suggest that swarms can be grouped according to design methods and analysis methods, as shown in figure 1.

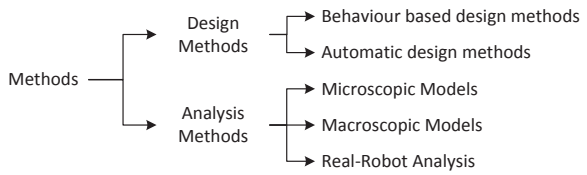

Fig. 1: Method Based Taxonomy [9]

Brambilla et al. claim that the most common design method is behaviour based, generally a bottom-up process, although it could be a top-down process, and is often based on the observations of animals. They categorise behaviour based designs into the three main categories: probabilistic finite state machine designs; virtual physics designs; and other design methods. They also suggest that automatic design methods are further classified as either evolutionary robotics or multi-robot reinforcement, and are employed with the aim of reducing the effort of developers as they do not require the intervention of the robotic swarm developer.

The authors also suggest that the analysis methods for robotic swarms can be modelled at either the individual level, the microscopic level, or the collective (macroscopic) level.

2.2 Collective Behaviour Based Taxonomy

In the same paper, Brambilla et al. present a second taxonomy, and categorise robotic swarms by their collective behaviours, as shown in figure 2.

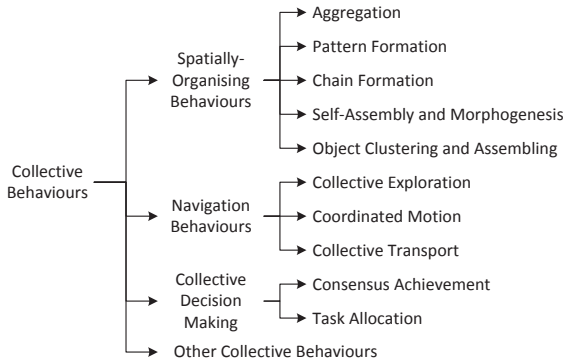

Fig. 2: Collective Behaviour Based Taxonomy [9]

The authors suggest classifying the collective behaviours by four main categories. These categories are based on how robotic swarms organise and distribute themselves; navigate and coordinate their movements; undertake decision making; and other collective behaviours that do not correspond to the above categories.

2.3 Swarm Behaviour Taxonomy

Cao et al. [21] also propose a taxonomy based on collective behaviour as shown in figure 3.

However, they choose different aspects to classify collective behaviour. In this taxonomy behaviour is categorised firstly by the system architecture. The remaining categories are based on: how resource conflicts are resolved; how cooperation between entities is achieved; how the swarm learns to solve problems; and how the collective entities conduct path planning from a geometric perspective.

2.4 Specification and Attributes Taxonomy

Another taxonomy has been proposed by Dudek et al. [12]. This classification scheme is based on a swarm's characteristics and attributes, as shown in figure 4.

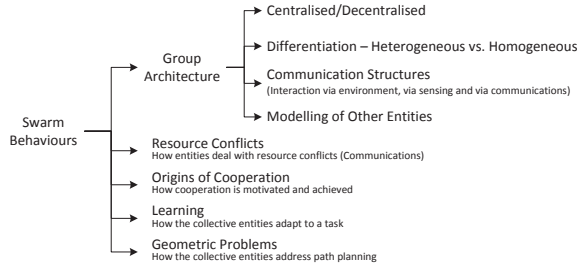

Fig. 3: Swarm Behaviour Taxonomy [21]

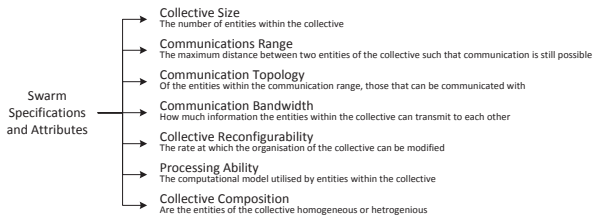

Fig. 4: Specification and Attributes Taxonomy [12]

Dudek et al. classify swarms based on the following categories. These are the size of the robotic swarm; the communications range between entities within a robotic swarm; the communication topology used; the quantity of information that can be transferred; how the swarm entities can re-configure their organisation; the processing capability that a swarm entity has; and whether the swarm's composition is homogeneous or heterogeneous.

3. Security Concepts

The above taxonomies give us a way to classify the unique aspects of robotic swarms where we wish to identify vulnerabilities. However, before we do so, we introduce some standard security terminology [23].

Confidentiality: Confidentiality is the requirement to protect information from unauthorised disclosure. Messages being transferred between elements of the swarm, might be confidential, as might data held by swarm elements.

Integrity: Integrity is the requirement to prevent data from being altered in an unauthorised way. This may also apply to messages between and data held by swarm elements.

Availability: This is the requirement for a system or data to be accessible and usable upon demand by an authorised entity. The term Denial of Service (DoS) is often used to describe a loss of availability.

Access Control: The requirement to allow only authorised access to a system. This generally requires identification. In a truly homogeneous swarm this may not be possible, although a group identity may be available.

4. The Adversary

We assume that the swarm will operate in a hostile environment. We also assume that the swarm may not always be in contact with its "owner". Thus the adversary may have access to the swarm without the "owner" being aware. These assumptions give a lot of power to the adversary but we believe they are reasonable since swarms are designed to operate autonomously and to be able to move freely.

Thus the adversary is able to freely observe and tamper with the swarm. In some cases, the adversary may be able to remove and replace elements of the swarm. Of course, any overt interference with the swarm is likely to be detected, but we assume our adversary wishes to operate covertly.

Our adversary's goal is to be able to manipulate the swarm. Ultimately the adversary would like to alter the emergent behaviour of the swarm.

5. Threats

Given an attacker with the capabilities described above we now consider the threats that are posed to swarm. In the following we use the term "threat" as defined by ISO/IEC 27005 [24]:

> "a potential cause of an incident, that may result in harm of systems and organisation"

This definition refers to the *potential* cause of an incident. So although many threats may exist, safeguards may be put in place to mitigate the risk of a threat being realised and exploiting vulnerabilities in the system. In order to cause harm, a threat needs to exploit a vulnerability of a asset within the system [25].

The following describes the threats to which any robotic swarm could be susceptible.

Denial of Service: The threat of a Denial of Service (DoS) is a threat to the availability of a system [26].

In addition to direct DoS threats to the system, a swarm is also subject to indirect threats via external environmental influences. For example, barriers placed by the adversary to impede free movement.

The DoS threat to a swarm will depend on the system design and where this design lies within the behaviour based taxonomies of Cao et al. [21] and Brambilla et al. [9]

Masquerade: Masquerade is the threat that an intruder or system entity illegitimately poses as, or assumes the identity of, another entity [27]. Masquerade allows unauthorised users to gain access to confidential data or greater privileges, while pretending to be legitimate user [28].

In terms of our adversary's goals, successful masquerade will allow covert operation.

System Penetration: The threat of system penetration is the unauthorised access to a system without posing as a legitimate entity [27]. This does not necessarily allow our adversary to operate covertly but, as with masquerade, it may enable the realisation of other threats.

Authorisation Violation: Authorisation violation is the use of a system by an *authorised* user for unauthorised uses, possibly abusing privileged access to resources. This can be undertaken by either an insider or external adversary [27].

In the theoretical model of a swarm, this threat may never be realised as there is no hierarchy within the swarm. However as noted by Cao et al. in their taxonomy, hierarchy may exist in practice. It is the "Group Architecture" branch of this taxonomy that may be subject to authorisation violation.

Planting: This is the threat that a malicious capability is planted within a system to perpetrate future attacks [27].

We are all familiar with the threats of Trojans and viruses. However, within a swarm, our adversary may be able to plant malicious *swarm entities* as described in section 4.

Eavesdropping: This threat is where an adversary obtains information by reading data sent between system participants, without penetrating the devices [27].

In our model, if messages are not protected, then our adversary could easily eavesdrop on the system without being detected.

Modification of Data in Transit: This is a threat to data in transit by either actively modifying the data or the communications process when sending data between authorised participants [27]. The threat could be to either or both user data and control information [29]. The threat is that an adversary can alter a legitimate message by deleting parts, adding extra, changing elements or reordering it [30].

Essentially this is a threat to integrity as defined in section 3 without penetrating a victim's system.

As with eavesdropping, if messages are not protected, our adversary could realise this threat without being detected.

Misappropriation: An attacker steals or makes unauthorised use of a service for which the service was initially unintended [30]. This is similar to authorisation violation but is not concerned with the escalation of privileges.

In our model, if this threat is realised, then the adversary could make a swarm execute a task for which it was not originally intended to undertake.

6. Attacks

An attack is the realisation of a threat [27]. We have considered a number of threats and how they relate to robotic swarms. Where appropriate, these threats will be realised by our adversary, and turned into attacks, to help achieve his goal of manipulating the swarm.

However, our adversary may only attack the *elements* within the swarm. The swarm, as a whole, is an abstract notion. Thus, individual swarm elements will be attacked, in order to manipulate behaviour of the swarm.

In other work [31] we proposed that there are four different types of attack that can be used to manipulate a swarm regardless of the swarm's specific implementation. These are summarised below.

Manipulate an element's goal: If attribute a_0 of a swarm element s_x defines it's goal, then manipulating a_0 (e.g. switch from "search" to "defend") can ultimately manipulate the swarm.

Manipulate an element's behaviour: A swarm element will have a number attributes, (e.g. separation, speed) If an adversary is aware of the effect that these attributes have on the behaviour of s_x, then he can, directly or indirectly, manipulate the attributes and ultimately manipulate the swarm.

Manipulate the environment: An adversary may alter the environment in order to manipulate the swarm.

Manipulate communications via the environment: The adversary may be able to modify the environment in order to inhibit communications or, in the case of stigmergy, to manipulate data in transit.

The above may be considered as generic attacks on a swarm. Since these are attacks on swarm elements there are numerous attack vectors that can be used. The specific attack vector used will depend on the technology used to implement the swarm, and to some extent, where the swarm lies in the Group Architecture class of Cao et al.

In the following, we list some of the more specific attack vectors that can be used against swarms. All of these attack vectors may be uses to implement one or more of the generic attack types described above.

- Replay: A replay attack is an attack in which a past message is played back to the same recipient [32]. In some cases it may be possible to modify parts of the message before replaying. Messages between swarm elements or from the environment may be replayed.
- Physical Tampering: We assume that an adversary may capture a swarm element. Having done so, the adversary my be able to physically attack the device to extract or modify data. This is different to traditional computing,

where physical protection of assets can be achieved by restricting physical access to devices.

- Software: Software or firmware attacks are concerned with modifying code, and exploiting vulnerabilities [33]. Software attacks on a swarm can be realised in several ways, such as via physical tampering, planting, or from within the supply chain.
- Communications: Attacks may be made on communications between swarm elements and between elements and the environment. Eavesdropping and modification of data are obvious attacks that can be made here. This becomes especially pertinent if the adversary is able to masquerade as a genuine swarm element, perhaps after capture and re-introduction to the swarm. Alternatively, the adversary could carry out these attacks from a static position in the area that the swarm is operating. An attacker may also block communications (DoS).
- Reconnaissance: In order to prepare for an attack, an adversary may undertake reconnaissance of a swarm in order to gain intelligence. This can be done by either traffic analysis or visual observation of the swarm.

7. Simulating an Attack

In order to test our hypothesis that we could manipulate a swarm by tampering with a few swarm elements we carried out some simulations using Netlogo 5.0.2.

7.1 Cooperative Navigation

We chose to simulate the "Cooperative Navigation" swarm proposed by Ducatelle et al. in 2013 [34]. The algorithm provided in this work was sufficient for us to design our simulation. The model was first simulated in a benign environment to ensure that our results were comparable.

The original research presented various options for cooperative navigation. This included a *searcher* attempting to find a *target* using information gained from other robots to aid in navigation.

From the *Method Based* taxonomy of Brambilla et al. this implementation would be considered as a *finite state machine* within the *Behaviour Based* design methods category. Many swarm implementations fall into this category.

In the *Collective Behaviour* based taxonomy of Brambilla et al. this would be classed as *Collective Exploration*, within *Navigational Behaviours*

In the *Group Architecture* class proposed by Cao et al. this activity would be classed as a *Communications Structure*.

Cao et al. may also consider this as a *Geometric Problem*, since the swarm is assisting in path planning. However, the collective entities are only passing on information to assist in path planning and not actually planning the path themselves.

In this swarm, the searcher and all entities contain stored information relating to *distance to the target* and the *information's age*. The target provides a count that is used to determine the age of the information received by each swarm element regarding it's distance to the target.

After a set time period, the target updates its count. Any entities from the swarm that are within the communication range of the target update their knowledge of their own distance to the target and the targets current count (for the information age). This data is then distributed through the swarm by sending messages to swarm members that are in range of each other. The receiving entity modifies its distance to the target if the information received is fresher than its stored information. The swarm does not act on this data, other than to forward the information on.

The searcher is also able to receive the distance to target and information age. If the data received by the searcher is "better" than its stored information then the searcher will update its data and the travel towards the position of the swarm entity that provided the information. By "better" we mean that either the information is fresher or if the information is of the same age but the distance to the target is shorter than the searcher's own data. The process continues until the searcher reaches the target.

If a searcher does not receive any information, or the swarm entities do not have any fresher information, then the searcher has two options. It can either stop and wait at its location for fresher information, or it can then travel in a random direction for a random distance, or until it receives fresher information.

7.2 Attacking Cooperative Navigation

Our initial attacks were Denial of Service. This was undertaken by the attacker manipulating the communications provided via the environment by means of jamming the signals. The jamming attacks are limited in both the frequency of the jamming event and the size of the area affected by the jamming event. The area affected was simulated to be the same size as the area that an individual robotic swarm entity could communicate within. Therefore communications were still possible between swarm entities that outside the jamming area.

Further experiments were then carried out where malicious entities manipulated the information being communicated between robotic swarm entities. This included manipulating distance to target and freshness of data values, in order to influence the robotic swarm entities.

7.3 Simulation Results

The effects of the DoS attacks can be seen in the figures 5 and 6 below. In these attacks, ten malicious entities are randomly positioned within the operating environment. The malicious entities, once dispersed, are static, have a 1% chance of undertaking a jamming event, and will jam for 5 time units.

Figure 5 shows how the effectiveness of an attack by various numbers of malicious entities, is reduced as the number

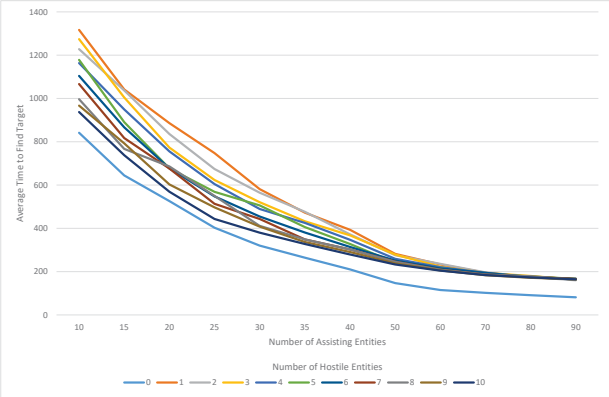

Fig. 5: DoS on a Robotic Swarm

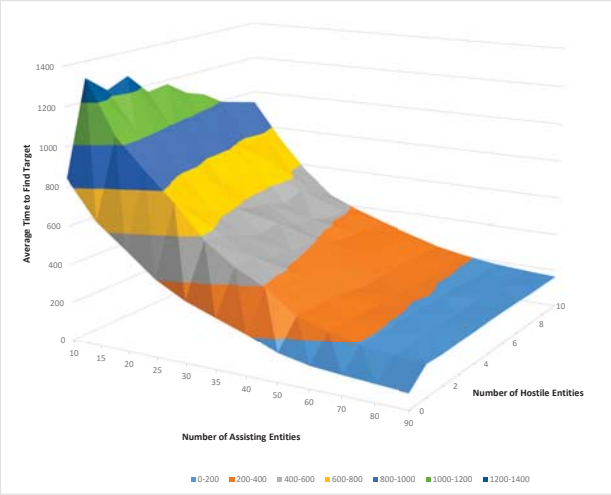

Fig. 6: Combined View of DoS on a Robotic Swarm

of genuine swarm entities increases. An attacker could use this information to optimise the number of malicious entities deployed for greatest effect.

Figure 6 demonstrates that this type of attack used can have unusual results. We designed the malicious entities to operate covertly, acting as part of the robotic swarm when not engaged in the DoS attack. Thus, when not attacking, the malicious entities actually assist the searcher in achieving its goal. Of course the adversary could counter this by increasing the attack period by increasing the likelihood of jamming events, or by operating a little less covertly. In the latter case, the malicious entities would monitor the swarm, and choose when to attack, but would not act as part of the robotic swarm.

7.4 Discussion

From our initial results we can see firstly, that we are able to manipulate the effectiveness of a swarm by introducing a small number of rogue elements. The simulation gives us a tool to study the effectiveness of an attack and investigate how to optimise the number of rogue elements.

Secondly, the simulation uncovers unexpected outcomes. Although, with hindsight, some outcomes are understandable, the non-deterministic nature of the swarm makes simulation a useful tool for both developer and adversary.

From this we can see that although the attack vectors are not new, their unique characteristics make swarms react in subtly different ways when compared with the same attack on traditional information systems.

The swarm chosen to simulate fits into several categories in the taxonomy. In theory we should now be able to choose swarms from similar categories and expect similar results. Alternatively we could investigate different classes of design.

We are currently running DoS simulations that utilise mobile jammers, in order to complete legacy style attacks. We are also running attacks based on the modification of data in transit. These attacks are aimed at tampering with data communications to manipulate the received distance to target and the information age. In other words, we are tampering with the attributes stored by individual swarm elements in order to manipulate the swarm.

8. Conclusion

In conclusion, this paper provides the researcher with an overview of various robotic swarm taxonomies and a understanding of various types of threats and attack methodologies. This is the basis for investigating vulnerabilities in different classes of robotic swarm systems.

From our initial experiments, we have demonstrated that robotic swarms can not only be attacked by traditional methods, but also that the attacks can utilise the unique characteristics of robotic swarm.

Our work is continuing to investigate the effectiveness of attacks and how the swarm may be able to defend itself against such attacks.

References

[1] P. Scerri, D. Pynadath, L. Johnson, P. Rosenbloom, M. Si, N. Schurr, and M. Tambe, "A prototype infrastructure for distributed robot-agent-person teams," in *Proceedings of the Second International Joint Conference on Autonomous Agents and Multiagent Systems*, ser. AAMAS '03. New York, NY, USA: ACM, 2003, pp. 433–440. [Online]. Available: http://www.cs.cmu.edu/~pscerri/papers/rap-aamas03.pdf

[2] S. Subchan, B. White, A. Tsourdos, M. Shanmugavel, and R. Zbikowski, "Pythagorean Hodograph (PH) Path Planning for Tracking Airborne Contaminant using Sensor Swarm," in *Instrumentation and Measurement Technology Conference Proceedings, 2008. IMTC 2008. IEEE*, 2008, pp. 501–506.

[3] H. Woern, M. Szymanski, and J. Seyfried, "The I-SWARM project," in *Robot and Human Interactive Communication, 2006. ROMAN 2006. The 15th IEEE International Symposium on*, Sept 2006, pp. 492–496.

[4] A. Martinoli, K. Easton, and W. Agassounon, "Modeling swarm robotic systems: A case study in collaborative distributed manipulation," *Int. Journal of Robotics Research*, vol. 23, pp. 415–436, 2004. [Online]. Available: http://www.kjerstin.com/pubs/04_AMKEWA_IJRR.pdf

[5] J. Fredslund and M. J. Mataric, "A general algorithm for robot formations using local sensing and minimal communication," *IEEE Transactions on Robotics and Automation*, vol. 18, no. 5, pp. 837–846, Oct 2002. [Online]. Available: http://robotics.usc.edu/publications/downloads/pub/47

[6] T. Balch and R. C. Arkin, "Behavior-based formation control for multi-robot teams," *IEEE Transactions on Robotics and Automation*, vol. 14, pp. 926–939, 1997. [Online]. Available: http://www.cc.gatech.edu/ai/robot-lab/online-publications/formjour.pdf

[7] C. M. Cianci, X. Raemy, J. Pugh, and A. Martinoli, *Swarm Robotics: Second International Workshop, SAB 2006, Rome, Italy, September 30-October 1, 2006, Revised Selected Papers*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, ch. Communication in a Swarm of Miniature Robots: The e-Puck as an Educational Tool for Swarm Robotics, pp. 103–115. [Online]. Available: http://infoscience.epfl.ch/record/100015/files/44330103.pdf

[8] B. P. Gerkey and M. J. Mataric, "Sold!: auction methods for multirobot coordination," *IEEE Transactions on Robotics and Automation*, vol. 18, no. 5, pp. 758–768, Oct 2002. [Online]. Available: http://cres.usc.edu/pubdb_html/files_upload/10.pdf

[9] M. Brambilla, E. Ferrante, M. Birattari, and M. Dorigo, "Swarm robotics: a review from the swarm engineering perspective," *Swarm Intelligence*, vol. 7, no. 1, pp. 1–41, 2013. [Online]. Available: http://dx.doi.org/10.1007/s11721-012-0075-2

[10] G. Dudek, M. R. Jenkin, E. Milios, and D. Wilkes, "A taxonomy for multi-agent robotics," *Autonomous Robots*, vol. 3, no. 4, pp. 375–397, 1996.

[11] G. Dudek, M. R. M. Jenkin, E. Milios, and D. Wilkes, "A taxonomy for multi-agent robotics," *Swarm Intelligence*, vol. 7, no. 1, pp. 1–41, 2013. [Online]. Available: http://dx.doi.org/10.1007/s11721-012-0075-2

[12] G. Dudek, M. Jenkin, and E. Milios, "A taxonomy of multirobot systems," *Robot Teams: From Diversity to Polymorphism*, pp. 3 – 22, 2002.

[13] G. Dudek, M. R. M. Jenkin, and D. Wilkes, "A taxonomy for swarm robots," in *Proceeding on the IEEE/RSJ International Conference on Intelligent Robotic Systems*. IEEE, 1993, pp. 441–447.

[14] E. Şahin, S. Girgin, L. Bayindir, and A. E. Turgut, "Swarm robotics," in *Swarm Intelligence: Introduction and Applications*, ser. Natural Computing Series, C. Blum and D. Merkle, Eds. Springer, 2008, pp. 87–100.

[15] A. F. T. Winfield and J. Nembrini, "Safety in Numbers: Fault Tolerance in Robot Swarms," *International Journal on Modelling Identification and Control*, vol. 1, no. 1, pp. 30–37, 2006.

[16] A. Yamashita, T. Arai, J. Ota, and H. Asama, "Motion planning of multiple mobile robots for cooperative manipulation and transportation," *Robotics and Automation, IEEE Transactions on*, vol. 19, no. 2, pp. 223–237, 2003.

[17] L. E. Parker, "Alliance: An architecture for fault tolerant multirobot cooperation," *Robotics and Automation, IEEE Transactions on*, vol. 14, no. 2, pp. 220–240, 1998.

[18] M. Gerla, Y. Yi, K. Xu, and X. Hong, "Team communications among airborne swarms," in *Aerospace Conference, 2003. Proceedings. 2003 IEEE*, vol. 3, March 2003, pp. 1303 – 1312.

[19] E. Chapman and F. Sahin, "Application of swarm intelligence to the mine detection problem," in *SMC (6)*, 2004, pp. 5429–5434.

[20] Y. C. Tan and B. Bishop, "Evaluation of robot swarm control methods for underwater mine countermeasures," in *System Theory, 2004. Proceedings of the Thirty-Sixth Southeastern Symposium on*, 2004, pp. 294–298.

[21] Y. Cao, A. Fukunaga, and A. Kahng, "Cooperative mobile robotics: Antecedents and directions," *Autonomous Robots*, vol. 4, no. 1, pp. 7–27, 1997. [Online]. Available: http://dx.doi.org/10.1023/A%3A1008855018923

[22] R. C. Arkin, T. Balch, and E. Nitz, "Communication of behavorial state in multi-agent retrieval tasks," in *Robotics and Automation, 1993. Proceedings., 1993 IEEE International Conference on*. IEEE, 1993, pp. 588–594.

[23] A. Menezes, P. van Oorschot, and S. Vanstone, *Handbook of Applied Cryptography*, ser. Discrete Mathematics and its Applications. Boca Raton, Fl: CRC Press, 1997, vol. 6. [Online]. Available: http://www.cacr.math.uwaterloo.ca/hac/

[24] *BS ISO/IEC 27005:2011 – Information technology – Security techniques – Information security risk management*, British Standards Std.

[25] *BS ISO/IEC 13335-1 Information Technology - Security Techniques - Management of Information and Communications Technology Security*, British Standards Std.

[26] M. Whitman and H. Mattord, *Principles of information security*. Cengage Learning, 2011.

[27] W. Ford and M. S. Baum, *Secure electronic commerce: building the infrastructure for digital signatures and encryption*. Prentice Hall PTR, 2000.

[28] W. Naik Bhukya, G. Suresh Kumar, and A. Negi, "A study of effectiveness in masquerade detection," in *TENCON 2006. 2006 IEEE Region 10 Conference*, Nov 2006, pp. 1–4.

[29] S. Kelly and T. C. Clancy, "Control and provisioning of wireless access points (CAPWAP) threat analysis for IEEE 802.11 deployments," 2009, Network Working Group, Internet Engineering Task Force.

[30] K. Scarfone, D. Dicoi, M. Sexton, and C. Tibbs, "Guide to securing legacy ieee 802.11 wireless networks," *NIST Special Publication*, vol. 800, p. 48, 2008.

[31] I. Sargeant and A. Tomlinson, "Review Of Potential Attacks On Robotic Swarms," in *IntelliSys 2016*. IEEE, 2016.

[32] L. Gong, "Variations on the themes of message freshness and replay-or the difficulty in devising formal methods to analyze cryptographic protocols," in *Computer Security Foundations Workshop VI, 1993. Proceedings*, Jun 1993, pp. 131–136.

[33] T. Roosta, S. Shieh, and S. Sastry, "taxonomy of security attacks in sensor networks and countermeasures," in *In The First IEEE International Conference on System Integration and Reliability Improvements. Hanoi*, 2006, pp. 13–15.

[34] F. Ducatelle, G. A. Caro, A. Förster, M. Bonani, M. Dorigo, S. Magnenat, F. Mondada, R. O'Grady, C. Pinciroli, P. Rétornaz, V. Trianni, and L. M. Gambardella, "Cooperative navigation in robotic swarms," *Swarm Intelligence*, vol. 8, no. 1, pp. 1–33, 2013. [Online]. Available: http://dx.doi.org/10.1007/s11721-013-0089-4